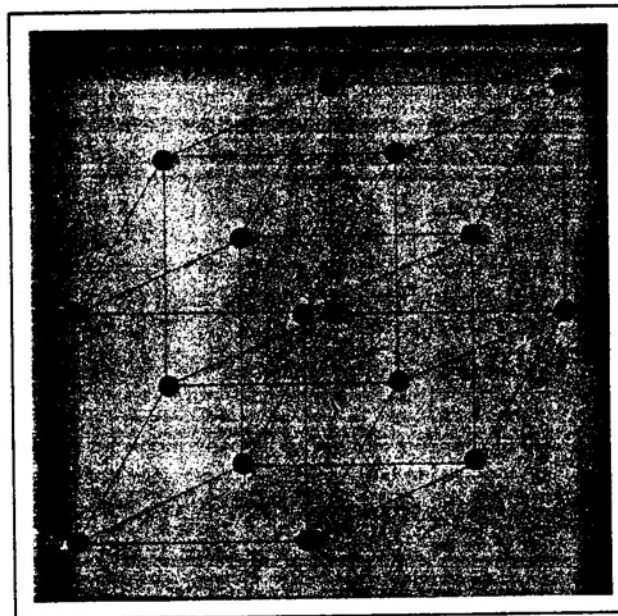


Integer Programming and Combinatorial Optimization

Proceedings of a Conference held at Carnegie Mellon University,
May 25-27, 1992, by the Mathematical Programming Society



Edited by
Egon Balas, Gérard Cornuéjols and Ravi Kannan

Algorithms and Min-max Theorems for certain Multiway Cuts

Péter L. Erdős

Dept. Applied Math., University of Twente, Enschede, The Netherlands, and
Mathematical Institute of the Hungarian Academy of Sciences

and

László A. Székely*

Institute of Operations Research, University of Bonn, and
Department of Computer Science, Eötvös University, Budapest

ABSTRACT: A greedy type algorithm and a min-max theorem is developed for the multiway cut problem of trees. Applications to biology also require complicated edge weights that also depend on the two components adjacent by the edge. We give a dynamic programming type algorithm for this more general problem and show that the min-max formula is no longer valid. As a substitute, we show a sort of min-max theorem that seems to fit the biological applications.

AMS classification codes (1991): 05C05, 05C70, 90C27

Key words: Menger's theorem, multiway cut, evolutionary tree

1. Introduction

Let $G = (V, E)$ be a simple graph, $C = \{1, 2, \dots, r\}$ be a set of colours. We denote by $E(G) \times 2$ the set of oriented edges of G . For $N \subseteq V(G)$, a map $\chi : N \rightarrow C$ is a *partial colouration*. A map $\bar{\chi} : V(G) \rightarrow C$ is a *colouration*, if $\chi(v) = \bar{\chi}(v)$ holds for all $v \in N$.

A map $W : E(G) \times 2 \rightarrow \mathbb{N}^{r \times r}$ is a (colouration dependent) *weight function* if the matrix $W(p, q)$ is the transpose of $W(q, p)$ and the main diagonal has only zero entries in every $W(p, q)$. The entry ${}_i W(p, q)_j = w(p, q; i, j)$ will tell the weight of the edge (p, q) in a colouration χ , in which $\chi(p) = i, \chi(q) = j$. We say that W is *colour independent*, if all the off-diagonal entries of the matrix $W(p, q)$ are the same. We say that W is *edge independent* if all oriented edges are mapped to the same matrix. We say that W is *constant*, if it is colour and edge independent.

An edge $\{p, q\}$ is *colour-changing* in the colouration $\bar{\chi}$, if $\bar{\chi}(p) \neq \bar{\chi}(q)$. The *changing*

* The second author was supported by the A. v. Humboldt-Stiftung

number of the colouration $\bar{\chi}$ is the sum of weights of the colour-changing edges in $\bar{\chi}$, i.e:

$$\text{change}(G, \bar{\chi}) = \sum_{\{p,q\} \in E(G)} w(p, q; \bar{\chi}(p), \bar{\chi}(q)).$$

A partial colouration χ defines a partition of N by $N_i = \{v \in N : \chi(v) = i\}$. A set of edges that separate every N_i from all the other N_j 's is termed a *multiway cut* [CR].

Observe that the set of colour-changing edges of a colouration $\bar{\chi}$ make a multiway cut and every multiway cut is represented in this way.

The *length* of the pair (G, χ) is the minimum weight of a multiway cut, in formula:

$$\ell(G, \chi) = \min\{\text{change}(G, \bar{\chi}) : \bar{\chi} \text{ colouration}\}.$$

An *optimal colouration* is a colouration $\bar{\chi}$ such that $\text{change}(G, \bar{\chi}) = \ell(G, \chi)$.

The *multiway cut problem for colour independent weight* has been extensively studied in combinatorial optimization (e.g. [CR], [DJPSY], [GH]). As Dahlhaus & al. pointed out [DJPSY], this problem is NP-hard even for $|N| = 3$, $|N_i| = 1$ and constant weight.

On the other hand, if we restrict ourselves to planar graphs and to a fixed number of colours and constant weight, then the problem becomes polynomially solvable [DJPSY]. We mention here, that a related problem, the minimum *k-partite cut* also has a polynomial time algorithm, see Goldschmidt and Hochbaum [GH]. The best known instance is $r = 2$, the undirected edge version of Menger's theorem [LP], which is polynomially solvable.

Although it is less known in the operations research community, some instances of the multiway cut problem have great importance in biomathematics. In fact, the notions of the changing number and the length came from genetics. Now assume that the weight is constant. An *evolutionary tree* is a leaf-coloured binary tree. The leaves represent species and their colours represent some genetical properties; and the tree is supposed to be the real tree of evolution of these species. For constant weight function Fitch [Fi] and Hartigan [Ha] developed polynomial time algorithm to determine the length of a given evolutionary tree.

The major problem is to find the most likely evolutionary tree for the given species and their colours. What biologists really would like to have, is the minimum length evolutionary tree over a set of species, in a much more complicated situation, where the colour of a species is a *word* over the alphabet C and *colourations* assign such words to branching vertices. Every bit on every edge of the tree may contribute by one to the changing number and the *length* of the tree is the minimum changing number over all colourations. A

common hypothesis is the *parsimony principle*, which says that the most likely evolutionary trees are those with minimum possible length. (For details see Felsenstein [Fe], Carter & al. [CHPSzW], Steel [St1].) To decide, if a tree with a given length (in this more general setting) exists, is NP-complete (see Graham and Foulds [GF]). Hence, we would like to have statistical information on the length as a tool to decide if an evolutionary tree is "acceptable" or not. Statistical information on evolutionary trees with one-letter colour-words in the leaves helps, see Steel [St1]; and in this paper we are concerned with one-letter colour-words only.

The statistical analysis of the most likely evolutionary trees requires the enumeration of evolutionary trees of length k built on a given, coloured leaf set. For $r = 2$ and $k = r-1$ the enumeration was done by Carter & al. [CHPSzW] using multivariate Lagrange inversion and computer algebra. Steel [St1] has found a combinatorial decomposition based on Menger's theorem to solve the enumeration problem for $r = 2$ and also solved the enumeration problem for $k = r$. Erdős and Székely [ES1] gave a rather simple presentation of Steel's decomposition, and understood that for the solution of the enumeration problem for arbitrary r and k it is necessary (but not sufficient) to come up with the proper generalization of Menger's theorem. It was the motivation to research for a min-max theorem on multiway cuts of evolutionary trees.

No matter how convenient is the assumption for constant weight on the edges of an evolutionary tree, it is not absolutely impeccable. Therefore Sankoff and Cedergren [SC], and Williamson and Fitch [WF] studied edge independent weight functions and made a polynomial time algorithm to find the length. Erdős and Székely [ES2] studied colour independent weight functions and developed a greedy-type algorithm to find the length.

In the present paper we provide a dynamic programming type algorithm to find the length of the tree equipped with an arbitrary weight function (Section 4), which seems to require higher complexity than the algorithms for edge or colour independent weight functions. Section 3 contains a min-max theorem for colour independent weight functions and its presentation in terms of linear programming. Section 5 shows that the min-max theorem does not generalize for arbitrary weight function, i.e. the polytope in question is not TDI. It also shows a sort of a weak min-max result, whose condition seems to be satisfied by "real" evolutionary trees.

2. Path packing

Let G be simple graph, $N \subseteq V(G)$ and $\chi : N \rightarrow C$ be a partial colouration. Let W be an

arbitrary weight function.

Definition. An oriented path P in G starting at $s(P) \in N$ and ending at $t(P) \in N$ is a *colour-changing path*, if $\chi(s(P)) \neq \chi(t(P))$ and P has no internal vertex in N . Let us fix a family $\mathcal{P}$ of colour-changing paths and let $e = (p, q)$ be an oriented edge of G . Define

$$n_i(e, \mathcal{P}) = \#\{P \in \mathcal{P} : (p, q) \in P \text{ and } \chi(t(P)) = i\}.$$

The notation $(p, q) \in P$ means that P enters the oriented edge (p, q) at p and leaves at q .

Definition. Let $\chi : N \rightarrow C$ be a partial colouration and $\bar{\chi}$ be a colouration on G . A family $\mathcal{P}$ of colour-changing paths is a *weak path packing* with respect to $\bar{\chi}$, if all the edges (p, q) for which $\bar{\chi}(p) \neq \bar{\chi}(q)$ satisfy

$$n_{\bar{\chi}(q)}((p, q), \mathcal{P}) + n_{\bar{\chi}(p)}((q, p), \mathcal{P}) \leq w(p, q; \bar{\chi}(p), \bar{\chi}(q)).$$

Definition. A colour-changing path system $\mathcal{P}$ is *strong path packing* if all pairs of colours colour $i \neq j$ and all oriented edges (p, q) satisfy

$$n_i((p, q), \mathcal{P}) + n_j((q, p), \mathcal{P}) \leq w(p, q; j, i).$$

The maximal cardinality of a strong path packing is denoted by $p(G, \chi)$.

Remark. Observe that any strong path packing is also weak path packing with respect to an arbitrary colouration.

Theorem 1. (i) Let χ be a partial colouration and $\bar{\chi}$ a colouration on the graph G . Let $\mathcal{P}$ be a weak path packing with respect to $\bar{\chi}$. Then

$$\text{change}(G, \bar{\chi}) \geq |\mathcal{P}|.$$

(ii) Let χ be a partial colouration, $\mathcal{P}$ be a strong path packing on G . Let $\bar{\chi}$ be an arbitrary colouration on G . Then

$$\text{change}(G, \bar{\chi}) \geq |\mathcal{P}|.$$

Proof. Due to the last Remark it is sufficient to prove (i). Let $\mathcal{P}$ be a weak path packing with respect to $\bar{\chi}$. Define a map $f : \mathcal{P} \rightarrow E(G)$ as follows: let $f(P) = e$ if e is the last colour-changing edge in P in $\bar{\chi}$. If $e = (p, q)$, $\bar{\chi}(p) = i$ and $\bar{\chi}(q) = j$ ($i \neq j$), then

$$\#\{P \in \mathcal{P} : f(P) = e\} \leq n_j(e, \mathcal{P}) + n_i(-e, \mathcal{P}) \leq w(p, q; i, j).$$

Therefore

$$|\mathcal{P}| \leq \text{change}(G, \bar{\chi}). \blacksquare$$

Remark. It follows from (ii) that for any graph G and partial colouration χ we have

$$\ell(G, \chi) \geq p(G, \chi).$$

We consider instances of G , for which the inequalities of Theorem 1 hold with equality.

3. Colour independent weight functions

This section is based on the papers [ES2] and [ES3]. Let G be a graph with a colour independent weight function $w : E(T) \rightarrow \mathbb{N}$ and with a partial colouration $\chi : N \rightarrow C$. Let us suppose that N intersects every cycle of G . Then we have an $O(nr)$ algorithm to determine the length of G and to find an optimal colouration. In fact, the algorithm has some wildcard steps, and making the proper choices at these steps, we may end up with an arbitrary optimal colouration. Furthermore, the following min-max theorem holds ([ES3]):

Theorem 2. *Let G be a graph with a colour independent weight function $w : E(G) \rightarrow \mathbb{N}$ and with a partial colouration $\chi : N \rightarrow C$. Let us suppose that N intersects every cycle of G . Then*

$$\ell(G, \chi) = p(G, \chi). \blacksquare$$

One may consider the following related linear programs:

For every oriented edge (p, q) of G and every ordered pair of distinct colours ij define a variable $z_{pq,ij}$. If $q \in N$, then eliminate $z_{pq,ij}$ and $z_{qp,ji}$ for every $j \neq \chi(q)$. Introduce new quotient variables by identifying the surviving variables $z_{pq,ij}$ and $z_{qp,ji}$ in pairs. For convenience we use the same notation for the quotient variables. Then the primal linear program is:

$$z_{pq,ij} \geq 0;$$

for every colour-changing path P_{ab} :

$$\sum_{(p,q) \in P_{ab}} \sum_{i: i \neq \chi(b)} z_{pq, i\chi(b)} \geq 1;$$

$$\min \sum z_{pq,ij} w(p, q),$$

where the sum is for all quotient variables. To describe the dual linear program, for every colour-changing path P_{ab} introduce a variable λ_{ab} , such that

$$\lambda_{ab} \geq 0;$$

for every quotient variable $z_{pq,ij}$:

$$\sum_{\substack{\chi(b)=j \\ (p,q) \in P_{ab}}} \lambda_{ab} + \sum_{\substack{\chi(v)=i \\ (q,p) \in P_{uv}}} \lambda_{uv} \leq w(p,q);$$

$$\max \sum \lambda_{ab}.$$

It is easy to see, that $p(G, \chi) \leq \max \sum \lambda_{ab} : \lambda_{ab} \text{ integer} \leq \max \sum \lambda_{ab} = \min \sum z_{pq,ij} w(p,q) \leq \min \sum z_{pq,ij} w(p,q) : z_{pq,ij} \text{ integer} \leq \ell(G, \chi)$. Only the first and last inequalities require proof from the chain of inequalities above. The first one holds, since any strong path packing provides a feasible integer solution for the second linear program. The last one holds, since we have an optimal colouration $\bar{\chi}$ with a total weight of colour-changing edges of $\ell(G, \chi)$; define $z_{pq,ij} = 1$, iff pq is a colour-changing edge in the optimal colouration $\bar{\chi}$ and $\bar{\chi}(p) = i$, $\bar{\chi}(q) = j$ hold, and $z_{pq,ij} = 0$ otherwise. If $\ell(G, \chi) = p(G, \chi)$, then equality holds everywhere in the chain.

4. An algorithm for arbitrary weight function

Let W be an arbitrary weight function. We describe our polynomial time algorithm to determine the length of a leaf-coloured tree and to find (all) optimal colourations.

Let T be a tree, let $L(T)$ denote the set of leaves, i.e. the set of vertices of degree 1. We assume $N = L(T)$. Let $B(T)$ denote the set of branching vertices $B(T) = V(T) \setminus L(T)$ and $b = |B(T)|$. In order to reduce the technical difficulties in describing the recursive algorithm we slightly modify the problem.

Definition. Let T be an arbitrary tree with a weight function $W : E(G) \times 2 \rightarrow \mathbb{N}^{r \times r}$. A *penalty function* is a vector-valued function $\text{pen} : L(T) \rightarrow \mathbb{N}^r$. Let $\bar{\chi} : V(T) \rightarrow C$ be an arbitrary colouration of T (as if we had $N = \emptyset$). The *changing number* of $\bar{\chi}$ on the edge (u, v) is

$$\text{change}((u, v), \bar{\chi}) = \begin{cases} w(u, v; \bar{\chi}(u), \bar{\chi}(v)) & \text{if } u, v \notin L(T) \\ w(u, v; \bar{\chi}(u), \bar{\chi}(v)) + \text{pen}_{\bar{\chi}(v)}(v) & \text{if } v \in L(T). \end{cases}$$

The *changing number* of the colouration $\bar{\chi}$ is

$$\text{change}(T, \bar{\chi}) = \sum_{e \in E(T)} \text{change}(e, \bar{\chi}).$$

Problem. Find a *minimal colouration* of T , i.e. one of minimum changing number.

Remark. If we have a partial colouration χ and K is a sufficiently large number, then we can trace back the optimal colouration problem to the minimal colouration problem as follows: If $\bar{\chi}(l) = i$ then set $\text{pen}_i(l) = 0$ and for every other colour j set $\text{pen}_j(l) = K$. Then a colouration of T is minimal iff it is optimal. Such a K is effectively computable, e.g. $K = \text{sum of all entries of all } W(p, q) \text{ matrices}$ is a right choice.

In order to determine all minimal colourations of T , we introduce the following:

Definition. A pair $X = (\overline{\text{pen}}; \rho)$ is an *extension* of the penalty function pen if $\overline{\text{pen}} : V(T) \rightarrow \mathbb{N}^r$, where $\overline{\text{pen}}(l) = \text{pen}(l)$ for all $l \in L(T)$, and the map $\rho : B(T) \rightarrow \{1, 2, \dots, b\}$ is a bijection, such that for every $k \leq b-1$ the vertex set $\rho^{-1}(\{b, b-1, \dots, b-k\})$ spans a subtree of T .

Definition. The colouration $\bar{\chi} : V(T) \rightarrow C$ is an *implementation* of the extension X if it can be constructed by the following algorithm:

Implementation algorithm

STEP 1.

Set w to be the maximum place of ρ . Set colour i to minimize $\overline{\text{pen}}_i(w)$. Set $A \leftarrow \{w\}$ and Set $\bar{\chi}(w) \leftarrow i$.

STEP 2.

While $B(T) \setminus A \neq \emptyset$

Set v to be the maximum place of $\rho|_{B(T) \setminus A}$. Set $u \in A$ to be the unique neighbour of v in A . Set colour i to minimize the value of $w(u, v; \bar{\chi}(u), i) + \overline{\text{pen}}_i(v)$. Set $\bar{\chi}(v) \leftarrow i$ and $A \leftarrow A \cup \{v\}$.

STEP 3.

For all $l \in L(T)$ let $v_l \in B(T)$ be the neighbour of l . Set colour i to minimize the value of $w(v_l, l; \bar{\chi}(v_l), i) + \overline{\text{pen}}_i(l)$. Set $\bar{\chi}(l) \leftarrow i$.

Definition. The extension X of the penalty function pen is *proper extension* if

- (i) every implementation of X is a minimal colouration,
- (ii) every minimal colouration is an implementation of X .

Theorem 3. Let T be a tree with penalty function pen and weight function W . Then there exists a proper extension X of pen .

Proof. We show a recursive algorithm to construct a proper extension X . A vertex $v \in B(T)$ is of *order one* if all of its neighbours but at most one are leaves. Let L_v denote the set of leaves adjacent to v . For every leaf $l \in L_v$ set

$$\bar{w}(v, l; j, i) := w(v, l; j, i) + \text{pen}_i(l).$$

Define the vectors $\min(v, l)$ for $l \in L_v$ by

$$\min_j(v, l) := \min_{i \in C} \bar{w}(v, l; j, i),$$

and finally set

$$\overline{\text{pen}}(v) := \sum_{l \in L_v} \min(v, l).$$

It is easy to see that if T is a star with midpoint v then this $\overline{\text{pen}}$ together with $\rho(v) = 1$ is a proper extension.

Let now T be a tree with $b > 1$ branching vertices. Take the truncated tree $T' := T \setminus L_v$ and keep the weight function on the surviving edges. For $l \in L(T')$ set

$$\text{pen}'(l) = \begin{cases} \text{pen}(l) & \text{if } l \in L(T) \cap L(T') \\ \overline{\text{pen}}(v) & \text{if } l = v. \end{cases}$$

For the tree T' we already have a proper extension due to the recursive procedure: $X' = (\overline{\text{pen}}', \rho')$. Now we are in a position to define $X = (\overline{\text{pen}}, \rho)$ for T :

$$\overline{\text{pen}}(u) = \begin{cases} \overline{\text{pen}}'(u) & \text{if } u \in L(T') \\ \text{pen}(v) & \text{if } u \in L_v \end{cases}$$

and

$$\rho(u) = \begin{cases} 1 & \text{if } u = v \\ \rho'(u) + 1 & \text{if } u \in B \setminus \{v\}. \end{cases}$$

Remark. One may prove easily that $\rho^{-1}(\{b, \dots, b-k\})$ is a connected subtree of T .

Claim. The extension $X = (\overline{\text{pen}}; \rho)$ is proper.

Proof. (A) Let $\bar{\chi} : V(T) \rightarrow C$ be an implementation of X . We have to show that $\bar{\chi}$ is an optimal colouration.

Then the map $\bar{\chi}'$ (which is just an abbreviation for $\bar{\chi}|_{T'}$) is an implementation of X' . Due to the recursive procedure it is a minimal colouration of T' . Let $\xi : V(T) \rightarrow C$ be an arbitrary colouration of T , and let $\xi' := \xi|_{V(T')}$. By the minimality of $\bar{\chi}'$

$$\text{change}(T', \bar{\chi}') \leq \text{change}(T', \xi').$$

Let ξ^* be the colouration of T obtained from ξ' by the implementation algorithm. Then

$$\begin{aligned} \text{change}(T, \bar{\chi}) &= \text{change}(T', \bar{\chi}') \leq \text{change}(T', \xi') \\ &\leq \text{change}(T', \xi^*) \leq \text{change}(T, \xi). \end{aligned}$$

(B) Assume $\bar{\chi} : V(T) \rightarrow C$ be a minimal colouration. We have to show that it is an implementation of the extension X . We show, that the colouration $\bar{\chi}' := \bar{\chi}|_{T'}$ is a minimal colouration.

Let $\xi : V(T) \rightarrow C$ be an arbitrary colouration of T , and let $\xi' := \xi|_{V(T')}$. Finally, let ξ^* be the colouration of T obtained from ξ' by the implementation algorithm. Then due to our algorithm

$$\text{change}(T, \xi^*) = \text{change}(T', \xi').$$

Finally, due to the optimality of $\bar{\chi}$ and the definition of $\overline{\text{pen}}$, we have

$$\text{change}(T', \bar{\chi}') = \text{change}(T, \bar{\chi}) \leq \text{change}(T, \xi^*) = \text{change}(T', \xi'),$$

therefore the colouration $\bar{\chi}'$ of T' is minimal. Using the hypothesis, $\bar{\chi}'$ is an implementation of X' . Now we can obtain $\bar{\chi}$ from $\bar{\chi}'$ by the implementation algorithm (this is just the basecase), consequently, $\bar{\chi}$ is an implementation of X . ■

Remark. The algorithm above determines a proper extension (and from it a minimal colouration) in polynomial time ($O(n^2 r^2)$). One may not expect to produce all minimal colourations in polynomial time, since M. A. Steel pointed out, that trees with constant weight may have exponentially many optimal colourations [St2].

5. A min-max formula that fails and a substitute

It is a natural question whether the packing linear program of Section 3 is totally dual integral, i.e. whether Theorem 2 generalizes for arbitrary weight functions. Unfortunately, this is not the case, take for example the 3-star with 3 differently coloured leaves, and the edge independent weight function

$$W = \begin{pmatrix} 0 & 1 & 3 \\ 1 & 3 & 0 \\ 3 & 0 & 1 \end{pmatrix}.$$

Then $\ell(G, \chi) = 4$, but $p(G, \chi) = 3$.

Definition. Let us be given a tree T and $\bar{\chi} : V(T) \rightarrow C$. An *elementary operation* is a contraction of a vertex of order one and one of its neighbours having the same colour; if the neighbour is a leaf, then the operation includes the deletion of the other leaf neighbours of that vertex. The colouration $\bar{\chi}$ is *recursive*, if iterated elementary operations can turn it into a star.

Theorem 4. Let W be an arbitrary weight function on the tree T , and let χ be a leaf-colouration of T . Then for every recursive colouration $\bar{\chi}$ there exists a weak coloured path packing $\mathcal{P}$ such that $\text{change}(T, \bar{\chi}) = |\mathcal{P}|$.

Proof. Let v be a vertex of order one (we use this definition and also the definitions of L_v and T' from the proof of Theorem 3). Let $\Sigma(T)$ denote the tree obtained from T by identification of the elements of L_v with identical colour and adding up their edge weights, respectively. Formally, denote T' the same subtree of T as in the proof of Theorem 3 and let the tree T^* be a star with midpoint v and with leaves $\{l_i : \exists u \in L_v : \chi(u) = i\}$. Finally, let $\Sigma(T)$ be the tree made of the trees T' and T^* by identification of their common v . Define the colouration and weight function of $\Sigma(T)$ as follows:

$$\bar{\chi}'(u) = \begin{cases} \bar{\chi}(u) & \text{if } u \in V(T') \\ i & \text{if } u = l_i, \end{cases}$$

$$W'(f) = \begin{cases} W(f) & \text{if } f \in E(T') \\ \sum_{\substack{u \in L_v \\ \chi(u)=i}} W((u, v)) & \text{if } f = (l_i, v). \end{cases}$$

Notice that $\text{change}(\Sigma(T), \chi') = \text{change}(T, \chi)$.

Claim. Let us suppose there is a weak path packing $\mathcal{P}'$ in the tree $\Sigma(T)$ such that $\text{change}(\Sigma(T), \bar{\chi}') = |\mathcal{P}'|$. Then there exists a weak path packing $\mathcal{P}$ in tree T such that $\text{change}(T, \bar{\chi}) = |\mathcal{P}|$.

Proof. Let L_v contain d different colours. We apply mathematical induction on $|L_v|$.

Base case: if $|L_v| = d$, then $\Sigma(T) = T$, $\bar{\chi} = \bar{\chi}'$, and we have nothing to prove.

Inductive step: Suppose that we know the Claim for all $|L_v| < k$. Assume now $|L_v| = k$ and for some fixed $z_1, z_2 \in L_v$, let $\bar{\chi}(z_1) = \bar{\chi}(z_2)$. Join z_1 and z_2 into z . In the new tree T^{**} obtained by identification, define the colouration and the weight function as follows:

$$\bar{\chi}^{**}(u) = \begin{cases} \bar{\chi}(u) & \text{if } u \neq z_1, z_2, \\ \bar{\chi}(z_1) & \text{if } u = z. \end{cases}$$

$$W^{**}(f) = \begin{cases} W(f) & \text{if } f \neq (v, z_i) \\ W(v, z_1) + W(v, z_2) & \text{if } f = (v, z). \end{cases}$$

Now we have $\Sigma(T) = \Sigma(T^{**})$, therefore

$$\text{change}(\Sigma(T), \bar{\chi}^*) = \text{change}(\Sigma(T^{**}), \bar{\chi}^{**}).$$

By hypothesis there exists a weak path packing $\mathcal{P}^{**}$ in the tree T^{**} such that $|\mathcal{P}^{**}| = \text{change}(T^{**}, \bar{\chi}^{**})$. It is easy to divide the paths of $\mathcal{P}^{**}$ adjacent to vertex z into two groups, such that the members of one group are adjacent to z_1 and the members of the other are adjacent to z_2 and both groups obey the weight restriction on the edge adjacent to z_i . In this way we obtain a weak path packing of $\text{change}(T, \bar{\chi})$ paths in T . ■

Due to the last Claim, we may suppose that for every vertex v of order one, every leaf of L_v has different colour in $\bar{\chi}$. Now one can prove Theorem 4 by induction on the cardinality $|B(T)|$.

Base case: Let T be a star with midpoint v . We distinguish two cases:

- (1) there is a leaf l such that $\bar{\chi}(v) = \bar{\chi}(l)$. Let us pack $w(v, u; \bar{\chi}(v), \bar{\chi}(u))$ many $l \rightarrow u$ paths for every leaf $u \neq l$. It is easy to check that this is the weak packing required.
- (2) No leaf has the colour $\bar{\chi}(v)$. Then for every leaf u we pack $w(v, u; \bar{\chi}(v), \bar{\chi}(u))$ many $l \rightarrow u$ paths, where l can be any other leaf (even more of them). It is also easy to check that this is the weak packing required.

Inductive step: Suppose that Theorem 4 holds for every tree T' with $b(T') < b(T)$. Since the colouration $\bar{\chi}$ is recursive, therefore there is a vertex v of order one and a neighbour of v , u , of the same colour, i.e. $\bar{\chi}(v) = \bar{\chi}(u)$. We distinguish two cases again:

- (1) Let us suppose that u is a leaf ($u \in L_v$). Let $T' = T \setminus L_v$, and apply the hypothesis for the tree T' . We have a weak path packing $\mathcal{P}'$ on T' of cardinality $\text{change}(T', \bar{\chi})$. Lift up the paths that end at v to u . Furthermore, apply the basecase for the star $T_v = v \cup L_v$ with the original colouration $\bar{\chi}$. We have a weak packing $\mathcal{P}^*$ of cardinality $\text{change}(T_v, \bar{\chi})$. It is easy to check, that the path system $\mathcal{P}' \cup \mathcal{P}^*$ is the weak path packing required.
- (2) Assume now, that u is not a leaf. Identify the vertices v and u and apply the hypothesis for the derived tree T' . We have a weak packing $\mathcal{P}'$ with cardinality $\text{change}(T', \bar{\chi})$. Lift-up $\mathcal{P}'$ into $\mathcal{P}$ to obtain a weak path packing in the original tree T . ■

Remark. We know from [ES3], that for a colour independent weight function all the optimal colourations are recursive, therefore Theorem 4 applies to them. (It does not give Theorem 2!) For an arbitrary weight function this is no longer valid. In fact, there is a tree on 6 vertices and 3 colours, such that none of the optimal colourations is recursive. Biologists will have a final word, if recursive colouration is a reasonable hypothesis or not. For us, it seems reasonable, since nature is economical with evolutionary changes.

Open problems: Find larger classes of graphs for which Theorem 2 holds. Find conditions for the weight function which imply equality in Theorem 1 (ii) for optimal coloura-

tions. What is the nature of Theorem 4 in terms of non-linear programming?

References

- [CHPSW] M. Carter, M. Hendy, D. Penny, L. A. Székely, N. C. Wormald, On the distribution of lengths of evolutionary trees, *SIAM Journal on Discrete Mathematics* 3 (1990), 38–47.
- [CR] S. Chopra, M. R. Rao, On the multiway cut polyhedron, *Networks* 21(1991), 51–89.
- [DJPSY] E. Dahlhaus, D. S. Johnson, C. H. Papadimitriou, P. Seymour, M. Yannakakis, The complexity of multiway cuts. Extended abstract (1983).
- [ES1] P. L. Erdős, L. A. Székely, Counting bichromatic evolutionary trees, to appear in *Discrete Appl. Math.*
- [ES2] P. L. Erdős, L. A. Székely, Evolutionary trees: an integer multicommodity max-flow–min-cut theorem, to appear in *Adv. in Appl. Math.*
- [ES3] P. L. Erdős, L. A. Székely, On weighted multiway cuts, submitted to *Mathematical Programming*.
- [Fi] W. M. Fitch, Towards defining the course of evolution. Minimum change for specific tree topology, *Syst. Zool.* 20 (1971), 406–416.
- [GF] R. L. Graham and L. R. Foulds, Unlikelihood that minimal phylogenies for a realistic biological study can be constructed in reasonable computational time, *Math. Biosci.* 60 (1982), 133–142.
- [GH] O. Goldschmidt, D. S. Hochbaum, A polynomial algorithm for the k -cut problem for fixed k , submitted to *JOR*.
- [GJ] M. R. Garey, D. S. Johnson, *Computers and Intractability: A Guide to the Theory of NP-Completeness*, W. H. Freeman and Co., New York, 1979.
- [Ha] J. A. Hartigan, Minimum mutation fits to a given tree, *Biometrics* 29 (1973), 53–65.
- [LP] L. Lovász, M. D. Plummer, *Matching Theory*, Akadémiai Kiadó, Budapest and North-Holland Publishing Co., Amsterdam, 1986.
- [SC] D. Sankoff, R. J. Cedergren, Simultaneous comparison of three or more sequences related by a tree, in: *Time Warps, String Edits, and Macromolecules* (Eds. D. Sankoff, J. B. Kruskal) Addison-Wesley, (1983).
- [St1] M. A. Steel, Distributions on bicoloured binary trees arising from the principle of parsimony, to appear in *Disc. Appl. Math.*
- [St2] M. A. Steel, Decompositions of leaf-coloured binary trees, to appear in *Adv. Appl. Math.*
- [WF] P. L. Williamson, W. M. Fitch, Finding the minimal changes in a given tree, in: *Trees and Hierarchical Structures, Lecture Notes in Biomathematics* No 84 (Eds. A. Dress, A. von Haeseler) Springer, (1988)